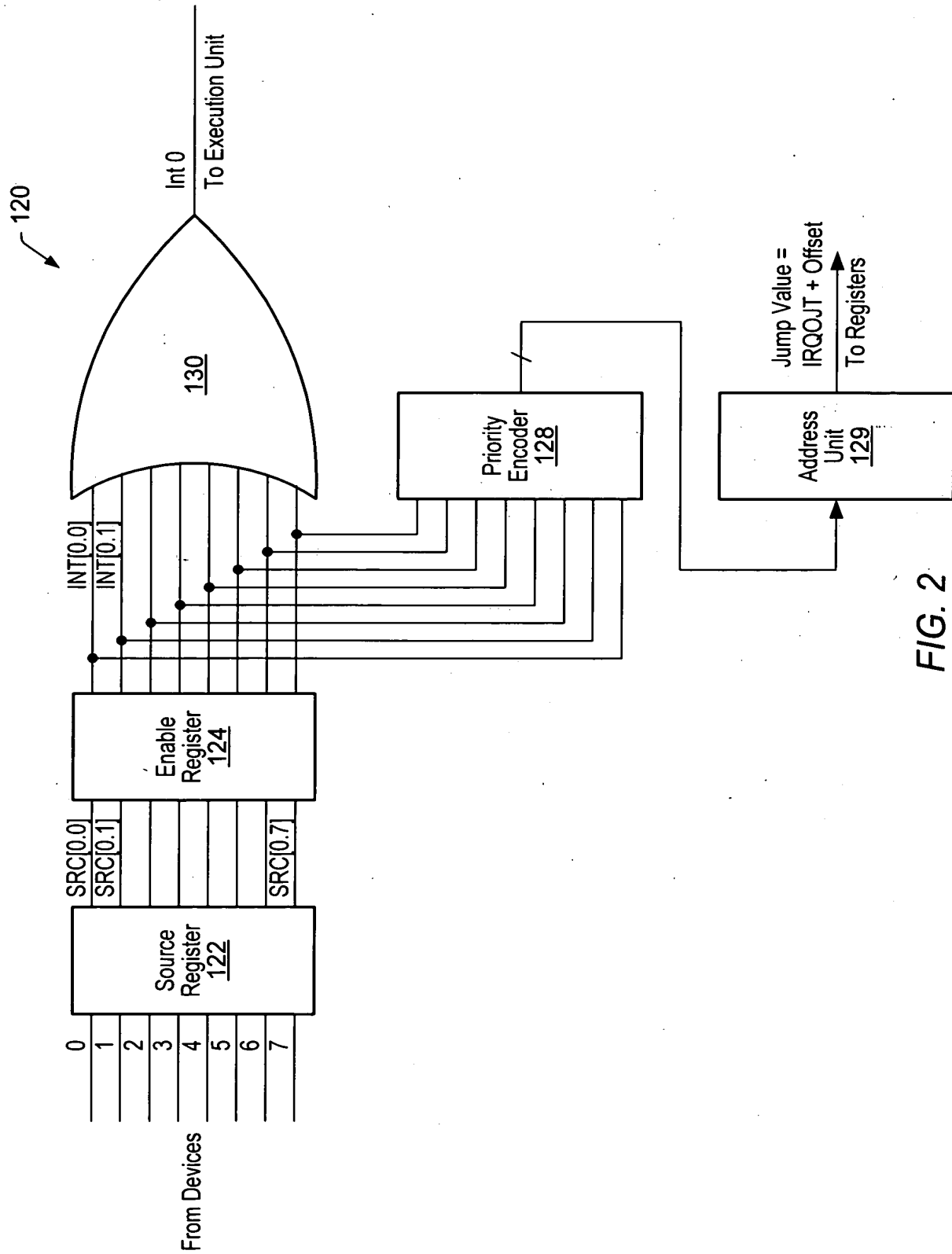


FIG. 1



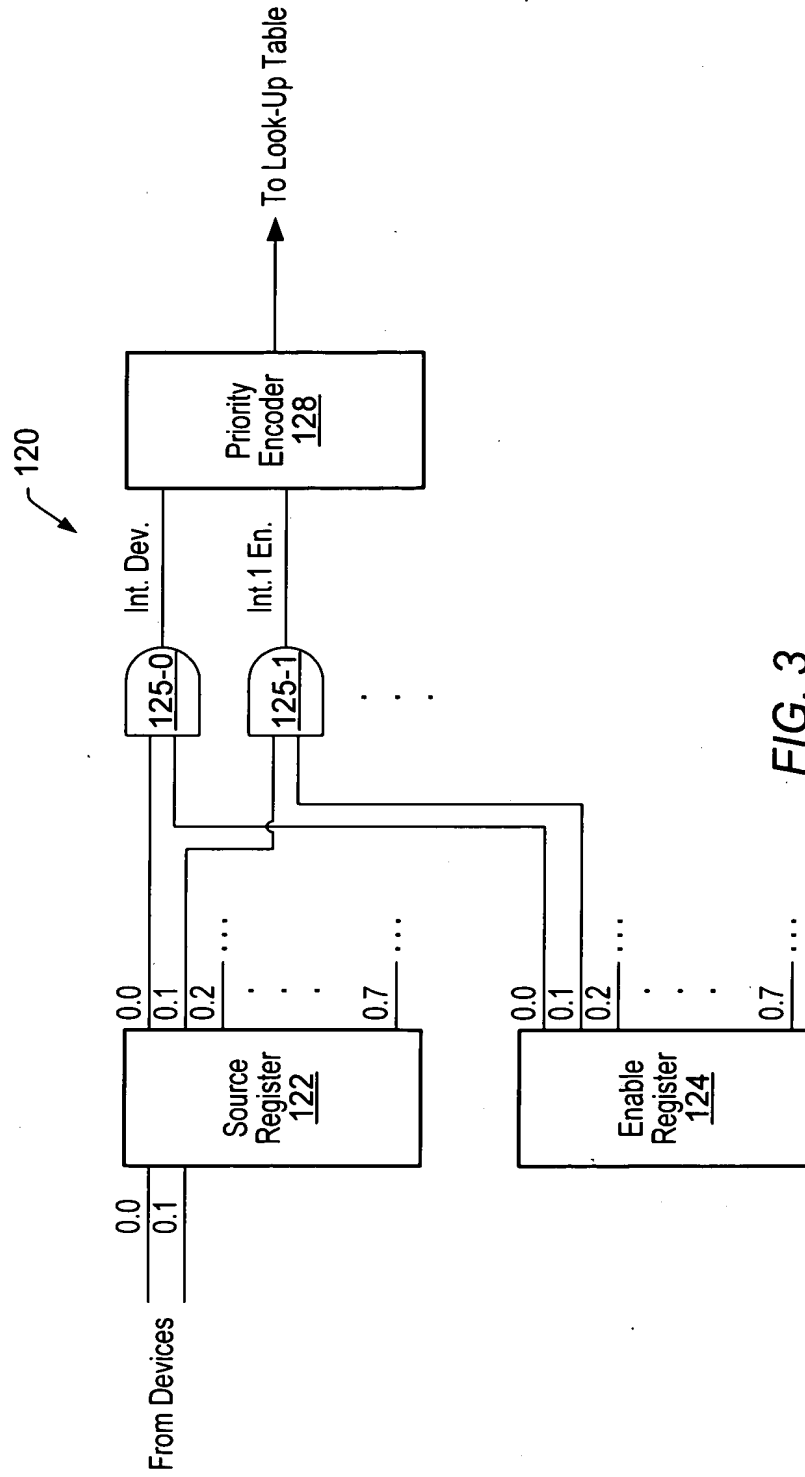


FIG. 3

405

| Addr | Instruction | Bytes | Cycles | |
|------|----------------------|-------|--------|--|
| 53 | LJMP (IRQ0JT+offset) | 3 | 4 | (IRQ4JT+offset) is set by the IRQ state machine based on the active sub-IRQ4 interrupt |
| 54 | - | | | |
| 55 | - | | | |
| 56 | - | | | |
| 57 | - | | | |
| 58 | - | | | |
| 59 | - | | | |
| 5A | - | - | - | |

410

| Addr | Instruction | Bytes | Cycles | Comment |
|--------|---------------|-------|--------|---|
| IRQ4JT | | | | IRQ4 Jump Table The location of this table is fixed by hardware and cannot be moved. |
| +00 | LJMP IRQ4.0 | 3 | 4 | IRQ4 Bit 0 service routine |
| +04 | LJMP IRQ4.1 | 3 | 4 | IRQ4 Bit 1 service routine |
| +08 | LJMP IRQ4.2 | 3 | 4 | IRQ4 Bit 2 service routine |
| +0C | LJMP IRQ4.3 | 3 | 4 | IRQ4 Bit 3 service routine |
| +10 | LJMP IRQ4.4 | 3 | 4 | IRQ4 Bit 4 service routine |
| +14 | LJMP IRQ4.5 | 3 | 4 | IRQ4 Bit 5 service routine |
| +18 | LJMP IRQ4.6 | 3 | 4 | IRQ4 Bit 6 service routine |
| +1C | LJMP IRQ4.7 | 3 | 4 | IRQ4 Bit 7 service routine |
| +20 | LJMP IRQ none | 3 | 4 | No IRQ4's are active-go to a null routine. This could have just a RETI or other processing. |

FIG. 4A

Modify ROM Code Table

| Addr | Instruction | Bytes | Cycles | |
|------|--------------|-------|--------|--|
| 53 | LJMP (EIRQ4) | 3 | 4 | (EIRQ4) is the entry into the extended Jump table. |
| 54 | - | | | |
| 55 | - | | | |
| 56 | - | | | |
| 57 | - | | | |
| 58 | - | | | |
| 59 | - | | | |
| 5A | - | - | - | |

| Addr | Instruction | Bytes | Cycles | Comment |
|-------|----------------------|-------|--------|--|
| | | | | The location of this table is fixed by hardware and cannot be moved. there is a table similar to this for each interrupt |
| EIRQ0 | LJMP (IRQ0JT+offset) | 3 | 4 | |
| EIRQ1 | LJMP (IRQ1JT+offset) | 3 | 4 | |
| EIRQ2 | LJMP (IRQ2JT+offset) | 3 | 4 | |
| EIRQ3 | LJMP (IRQ3JT+offset) | 3 | 4 | |
| EIRQ4 | LJMP (IRQ4JT+offset) | 3 | 4 | Jump to offset into IRQ4JT |
| EIRQ5 | LJMP (IRQ5JT+offset) | 3 | 4 | |
| EIRQ6 | LJMP (IRQ6JT+offset) | 3 | 4 | |
| EIRQ7 | LJMP (IRQ7JT+offset) | 3 | 4 | |
| EIRQN | LJMP (EIRQN+offset) | 3 | 4 | No interrupt is active, jump to a null routine. |
| | | | | |
| | | | | |

FIG. 4B

| Addr | Instruction | Bytes | Cycles | Comment |
|-------|---------------|-------|--------|--|
| EIRQ0 | | | | IRQ4 Jump Table The location of this table is fixed by hardware and cannot be moved. There is a table similar to this for each interrupt. |
| +00 | LJMP IRQ4.0 | 3 | 4 | IRQ4 Bit 0 service routine |
| +04 | LJMP IRQ4.1 | 3 | 4 | IRQ4 Bit 1 service routine |
| +08 | LJMP IRQ4.2 | 3 | 4 | IRQ4 Bit 2 service routine |
| +0C | LJMP IRQ4.3 | 3 | 4 | IRQ4 Bit 3 service routine |
| +10 | LJMP IRQ4.4 | 3 | 4 | IRQ4 Bit 4 service routine |
| +14 | LJMP IRQ4.5 | 3 | 4 | IRQ4 Bit 5 service routine |
| +18 | LJMP IRQ4.6 | 3 | 4 | IRQ4 Bit 6 service routine |
| +1C | LJMP IRQ4.7 | 3 | 4 | IRQ4 Bit 7 service routine |
| +20 | LJMP IRQ none | 3 | 4 | No IRQ4's are active-go to a null routine. This could have just a RETI or other processing. |
| | | | | |
| | | | | |

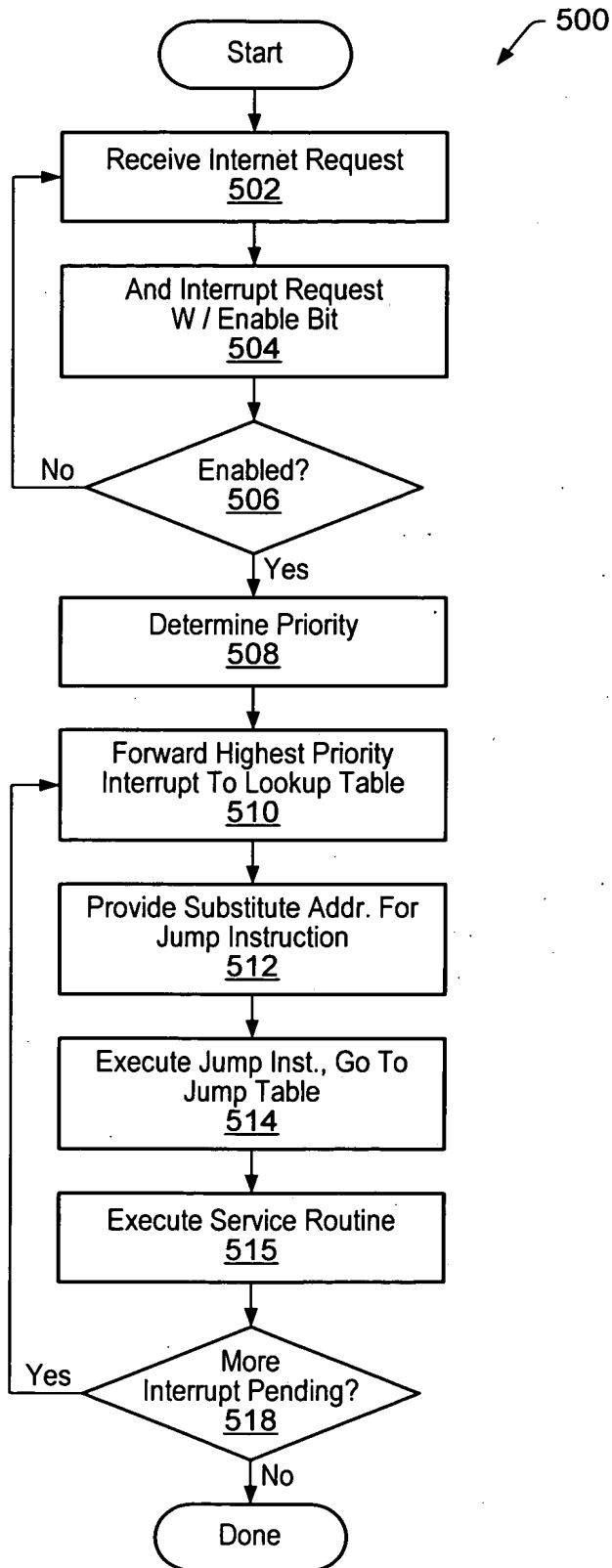


FIG. 5